Paper

Manipulation of compressed data using MPEG-7 low level audio descriptors

Jason Lukasiak, David Stirling, Shane Perrow, and Nick Harders

Abstract — This paper analyses the consistency of a set of MPEG-7 low level audio descriptors when the input audio stream has previously been compressed with a lossy compression algorithm. The analysis results show that lossy compression has a detrimental effect on the integrity of practical search and retrieval schemes that utilize the low level audio descriptors. Methods are then proposed to reduce the detrimental effects of compression in searching schemes. These proposed methods include improved searches, switched adaptive scalar and vector prediction, and other prediction schemes based on machine learning principles. Of the proposed schemes the results indicate that searching which incorporates previous and future frames combined with machine learning based prediction best nullifies the effects of compression. However, future scope is identified to further improve the reliability of the MPEG-7 audio descriptors in practical search environments.

Keywords — MPEG-7, metadata, multimedia description, machine learning, multimedia retrieval.

1. Introduction

With the ever increasing volume of multi-media (MM) data available via shared networks, such as the Internet or even large organizational intranets, meaningful and efficient storage, retrieval, archiving and filtering of the available MM data is becoming increasingly difficult. Current text based search and retrieval schemes rely on meaningful textual tags being associated with every MM item. Such text based methods are limited in usefulness by the quality (content) of the tag used. For example, it may be possible to find a particular MM item via Authors name or title, but it is extremely unlikely that content specific features such as colour, melody or frequency structure would be identifiable using a text based tag. Overcoming this limitation is the realm of the new MPEG-7 standard [1]. This standard provides a structured framework for describing MM content in a platform independent environment [1, 2].

At its lowest level, the MPEG-7 standard specifies a set of low level descriptors that are calculated directly from the MM content [2]. Examples of the low level descriptors generated are colour space [2] and audio spectral envelope [3]. A detailed description of the entire MPEG-7 standard can be found in [1] and an excellent overview in [2].

Having descriptors associated with MM data that describe the actual content of the data, provides the potential for powerful manipulation of content supply and consumption. The manipulation could involve finding all MM items in a database that have a blue background or selecting the audio segments that represent specific sources (such as dogs barking) [4, 8]. Such searching could feasibly return all of the images in a database that contain "Sampras playing tennis" [5].

Whilst the proposed MPEG-7 standard offers a powerful new scheme for control of MM data, there are a number of issues that may limit practical application of the standard. Examples of such limitations are the complexity involved and the integrity of the descriptors in compressed environments. As a substantial amount of MM data is compressed before storage using various lossy compression schemes, such as MP3 for audio and JPEG for images, the perceptually redundant information from the signal is substantially removed. Consequently, the effect of compression on the integrity of the descriptors is extremely important for practical applications. For example, can we find an MP3 audio file (or an audio segment previously compressed by MP3) that matches our target song, using the low level descriptors generated from a CD?

The effect of compression on the MPEG-7 low level audio descriptors is the focus of this paper. A thorough analysis into the effect of compression on five of the seventeen low level audio descriptors is presented in Section 2. These five descriptors were selected due to space constraints in presenting results for the full set of descriptors, and also, due to these descriptors being frame based (as opposed to entire file based). The combination of the prescribed descriptors also presents a compact description of the underlying audio data.

Once identified some preliminary methods for reducing the effects of compression on the low level descriptors are detailed in Section 3. Finally the major points are summarized in Section 4.

2. Effect of compression on low level descriptors

The five audio low level descriptors selected for the analysis were: audio power (AP), audio waveform (AW), audio spectrum ennvelope (ASE), audio spectrum centroid (ASC)

and audio spectral spread (ASS). A full description of these can be found in [3].

To determine the effect of compression on the audio low level descriptors, two well known audio compression algorithms, MP3 [6] and WMA [7], were used to compress and uncompress 90 16-bit 44.1 kHz sampled audio files. Each of the audio files was of approximately 10 seconds duration, with 50 files representing instrumental only signals (inst) and 40 representing combinational signals (comb) such as pop music.

The MP3 encoder was operated at 128, 160 and 192 kbit/s and the WMA coder was operated at both 128 and 160 kbit/s.

The MPEG-7 low level audio descriptors defined above, were then calculated for both the files which had been compressed/uncompressed and the original files. A frame size of 20 ms was used to calculate the descriptors.

2.1. Objective measures

To give an objective indication of the effect of compression on the descriptors, the segmented signal to noise ratio (SegSNR) was used. This was calculated as:

$$\text{SegSNR} = \frac{1}{N}\sum_{x=0}^{N-1} 10\,\log_{10}\left(\frac{\sum_{n=1}^{r} x_n^2}{\sum_{n=1}^{r}(x_n-\overline{x_n})^2}\right), \qquad (1)$$

where x_n is the descriptor from the original file, $\overline{x_n}$ is the descriptor from the compressed file, r is the dimension of the descriptor per frame and N is the number of frames in the file. The SegSNR for each descriptor and compression configuration was calculated for each file, with the results summarized in Sections 2.3–2.8.

2.2. Practical measures

Determining the effect of compression noise on the descriptors requires not only objective measures, but also analysis of the performance in a practical searching scheme. To this end, a simple searching scheme was utilized that attempts to locate a specific frame in the compressed file, using the descriptors generated for the target frame from uncompressed data. This is a simplified version of search/retrieval schemes outlined in [8, 9]. The searching scheme selects the frame in the compressed file that minimizes the mean squared error (MSE) defined as:

$$\text{MSE} = \frac{1}{r}\sum_{n=1}^{r}(x_n-\overline{x_n})^2, \qquad (2)$$

where x_n, $\overline{x_n}$ and r are as defined for Eq. (1). It should be noted that when the descriptors are generated from original uncompressed data, the above scheme returns the correct frame for all individual (and combinations of) descriptors.

2.3. Results for AP

The AP describes the instantaneous power of each input frame [3]. AP consists of only a single scalar value per frame. The average SegSNR values for the instrumental files, combinational files and the overall average for all files, for each compression scheme, are shown in Table 1.

Table 1
SegSNR values for AP [dB]

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	33.84	36.75	37.68	36.12	38.53
Inst.	40.13	43.2	44.46	42.3	44.78
Average	37.33	40.33	41.44	39.55	42

The results in Table 1 indicate that the AP descriptor achieves a very high SegSNR value of over 40 dB for all of the compression schemes. This high value would indicate that compression noise has little effect on the value of the descriptor.

It is interesting to note that there is approximately a 5–7 dB improvement in SegSNR for the instrumental files when compared to the combination files. This distinct difference is most likely attributed to the increased masking effect present in the spectrally rich combinational files. This increased masking allows the compression schemes to remove more redundant information from the combinational files than the instrumental files, and thus, the objective difference between the compressed and original files is greater for the combinational files.

The average search results achieved when using only the AP to identify the target frame are shown in Table 2. The results in Table 2 indicate the percentage of incorrect frames identified for the first 5 instrumental and combination files.

Table 2
Percentage incorrect frames for AP search

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	84.3	84.1	86.8	80.6	75.7
Inst.	85.2	84.5	85.2	84.3	81.1
Average	84.75	84.3	86	82.45	78.4

The results in Table 2 indicate that despite the very high SegSNR values reported in Table 1, the AP descriptor is a very unreliable search mechanism. This inability to adequately match the target frame is due to both the AP descriptor having very similar values across adjacent frames and the fact that the logarithmic amplitude quantisation is employed in most audio encoders.

2.4. Results for AW

The AW descriptor provides a low resolution representation of the time domain envelope [3]. It consists of 2 scalar values (max and min) per frame. The average SegSNR values for the same files used in Table 1 are shown in Table 3.

Table 3
SegSNR values for AW [dB]

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	39.51	39.57	38.83	41.49	44.36
Inst.	45.28	46.15	46.55	47	49.83
Average	42.71	43.22	43.12	44.55	47.4

The results in Table 3 indicate that whilst the AW descriptor has slightly lower SegSNR values than those for AP shown in Table 1, the values are still very high. The worst case is 33.84 dB for MP3 using the combination files.
The average search results achieved using the AW descriptor for the same files used in Table 2 are shown in Table 4.

Table 4
Percentage incorrect frames for AW search

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	50.4	40.7	35.1	40.5	30.8
Inst.	62.1	52.1	47.5	55.1	46.3
Average	56.25	46.4	41.3	47.8	38.55

The results presented in Table 4 indicate that despite the AW exhibiting lower SegSNR values than the AP descriptor, it provides a more reliable searching mechanism. This improvement is due to the fact that the AW descriptor has two values for each frame and the probability of two frames having very similar descriptors is reduced. However, the AW descriptor still finds incorrect frames on approximately 40 to 60 percent of occasions.

2.5. Results for ASC

The ASC descriptor represents the center of gravity of the frequency spectrum [3]. The descriptor is a single scalar value per frame that indicates the octave shift from 1 kHz of the centroid value. The average SegSNR values for the same files used in Table 1, are shown in Table 5.
The results in Table 5 indicate that in an objective sense, compression has very little effect on the ASC descriptors. This result is clearly evidenced by the smallest value for SegSNR being in excess of 40 dB.

Table 5
SegSNR values for ASC [dB]

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	41.09	44.06	44.29	44	46.81
Inst.	47.46	50.64	51.96	50.79	53.51
Average	44.63	47.72	48.55	47.77	50.53

The average search results achieved using the ASC descriptor for the same files used in Table 2 are shown in Table 6.

Table 6
Percentage incorrect frames for ASC search

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	84.6	77.6	76.4	78.2	77.6
Inst.	88.5	87.4	85.9	88.1	87.4
Average	86.55	82.5	81.15	83.15	82.5

The results in Table 6 indicate that due to the relative stability of the centroid value across frames, the minor effects of compression evident in Table 5 cause the ASC descriptor to be unsuitable for frame identification in compressed environments.

2.6. Results for ASS

The ASS descriptor describes the RMS deviation from the centroid value (ASC) for a given frame [3]. The descriptor consists of a single scalar value per frame that represents the octave spread from the ASC value. The average SegSNR values for the same files used in Table 1 are shown in Table 7 and the average search results achieved using the ASS descriptor for the same files used in Table 2, are shown in Table 8.

Table 7
SegSNR values for ASS [dB]

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	45.47	47.17	47.11	48.21	51.06
Inst.	49.01	50.76	52.09	52.82	55.8
Average	47.43	49.15	49.87	50.77	53.69

As for the ASC descriptor, despite achieving high SegSNR values (as shown in Table 7) the ASS provides a very unreliable search mechanism in compressed environments.

Table 8
Percentage incorrect frames for ASS search

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	88.6	79.2	87.5	83.6	79.2
Inst.	92.5	92.8	92.1	88.1	87.5
Average	90.55	86	89.8	85.85	83.35

The descriptor often finds the incorrect frame in over 90% of instances, as shown in Table 8. This poor search performance is again attributed to the fact that the ASS values may vary only marginally between frames, and hence, the relatively small amount of compression noise introduced is sufficient to cause ambiguity when searching for absolute matches to the ASS values.

2.7. Results for ASE

The ASE descriptor provides a representation of the power spectrum for each frame of the audio file. The descriptor consists of a vector of values for each frame, with each vector component representing the magnitude of a particular frequency band. The number of frequency bands (and hence the length of the ASE descriptor) is variable according to a predetermined set of user parameters. These parameters include loEdge, hiEdge and resolution [3]; where loEdge represents the lowest edge (frequency) of the frequency bands, hiEdge represents the highest edge (frequency) of the frequency bands and resolution defines the width of the frequency bands (in octaves with respect to 1 kHz) between loEdge and hiEdge. The ASE also contains two additional values representing 0 Hz-loEdge and hiEdge-Sampling_freq/2.

An important note in the standard [3], is that for fine resolutions (i.e. $< \frac{1}{4}$ octave) the window length (length of frame) restricts the minimum value for loEdge such that at least 1 FFT frequency coefficient is present in each band. The result of this restriction is that for resolutions less than $\frac{1}{4}$ octave, the resultant ASE descriptor becomes biased. This bias is due to the fact that the value representing 0 Hz-loEdge has many FFT coefficients lumped into it, and thus, becomes very large, whilst the neighbouring bands contain only a single coefficient. The net result of this effect is that resolutions less than $\frac{1}{4}$ octave are not searching or identifying complex audio signals that have significant low frequency content (such as speech). To alleviate this problem the ASE descriptor generated for this work used a resolution of $\frac{1}{4}$ octave, which produced 32 frequency bands for each frame.

The average SegSNR values for the same files used in Table 1 are shown in Table 9.

The values in Table 9 indicate that the ASE descriptor produces lower objective results in the presence of compression than the other spectral descriptors, ASS and ASC.

Table 9
SegSNR values for ASE [dB]

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	31.33	33.85	37.49	33.88	36.63
Inst.	38.37	42.17	43.66	40.55	43.32
Average	35.24	39.05	40.36	37.58	40.35

This should be expected as the ASE produces much finer resolution than those other descriptors, and hence, the effects of removing masked components in the compression scheme produces more visible objective distortions. It is also clearly evident that as the bit rate of the compression schemes increases, the SegSNR also increases. This effect is due to the compression schemes using the additional bits available to better represent the spectral envelope of the signal.

The average search results achieved using the ASE descriptor for the same files used in Table 2 are shown in Table 10.

Table 10
Percentage incorrect frames for ASE search

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	0.15	0.15	0.18	0.18	0.18
Inst.	2.5	0.8	1	2.5	1.9
Average	1.325	0.475	0.59	1.34	1.04

The results in Table 10 indicate that the ASE produces fairly reliable search results for all compression schemes. This is obviously due to the fine resolution present in the descriptor and thus the likelihood of two similar frames existing (even in the presence of compression noise) is lower than for the more generic descriptors. However, the search still fails approximately 2% of the time for the instrumental files.

2.8. Results for combined descriptor searches

To ascertain if improved search results could be achieved by combining multiple descriptors together into meta-descriptors, we formed two meta-descriptors. The first of these meta-descriptors combined all five of the specified descriptors together and the second combined ASC and ASS to produce a compact representation of frequency content. The search results for these two meta-descriptors using the same files used in Table 2 are shown in Tables 11 and 12 respectively.

Comparing the results in Table 11 to those for the ASE only in Table 10 indicates that including the additional descriptors into the search actually degrades the searching

Table 11
Percentage incorrect frames for meta-descriptor using all of the original descriptors

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	1.8	0.74	0.3	0.5	0.1
Inst.	16.3	8.7	4.8	7.8	4.4
Average	9.05	4.72	2.55	4.15	2.25

Table 12
Percentage incorrect frames for the ASS/ASC meta-descriptor

	MP3			WMA	
Bit rate [kbit/s]	128	160	192	128	160
Comb.	24.5	16.2	13.2	13.5	8.8
Inst.	44.7	37	31.2	34.2	24.1
Average	34.6	26.6	22.2	23.85	16.45

performance. This result indicates, that because the additional descriptors may have larger absolute values than the individual ASE components, the ambiguity introduced into these additional descriptors by compression is sufficient to degrade the unweighted search performance used in Eq. (2). Better results may be achieved by introducing a weighting function into Eq. (2).

When compared to the results for the descriptors ASC and ASS in isolation (Tables 6 and 8 respectively), the results in Table 12 indicate that combining the descriptors together reduces the incorrect results by between 50 and 66%. This improved result is encouraging and supports the finding that implementing more sophisticated search (weighted) mechanisms for meta-descriptors may improve performance in compressed environments.

2.9. Summary of results

The results presented for all descriptors indicate that an objective measure of the effects of compression gives little indication of the actual performance degradation in a practical search situation. The results indicate that compression noise is a significant problem for practical applications of the low level audio descriptors.

The primary reason for the modification of the MPEG-7 audio descriptors when using compressed input data, is the redundancy removal performed by the compression algorithms. For audio compression, redundancy is removed (compression achieved) by hiding quantisation noise in sections of the spectrum that are masked (inaudible to the human ear). This procedure may result in the actual spectral shape being significantly modified by the compression algorithm. As many of the frame based MPEG-7 low level audio descriptors are based on representations of the spectrum, this modification of the spectral shape directly affects the values of the MPEG-7 descriptors calculated from the compressed input stream. Also, as most audio compression schemes quantise amplitude values on a logarithmic scale that replicates the response of the human ear, the linear representation of the audio power provided by the AP descriptor suffers from quite severe quantisation noise for large amplitudes. These effects in combination with the fact that many of the descriptors analysed vary only marginally between frames, cause the presented MPEG-7 audio descriptors to produce very unreliable searching parameters.

It should be noted that on average, across all of the results presented in Sections 2.3–2.8, the WMA coder had less effect on the descriptors than the MP3 encoder. This result is most likely attributed to the fact that the WMA algorithm is significantly more modern than MP3, and thus, exploits more sophisticated signal processing and psycho-acoustic techniques.

3. Methods for improving performance in compressed environments

The methods examined for improving the search performance in compressed environments can be grouped into two categories: 1) adaptive signal processing (ASP) and 2) machine learning.

3.1. ASP techniques

A number of signal processing techniques were examined. The first of these was a simple fixed predictor that attempts to predict the original descriptors from the compressed descriptors. The predictor coefficient was calculated as:

$$a = \frac{\sum_{n=0}^{N-1} x(n)\overline{x(n)}}{\sum_{n=0}^{N-1} x(n)^2}, \tag{3}$$

where a is the predictor coefficient and x_n, $\overline{x_n}$ and N are as defined for Eq. (1).

The more sophisticated technique of vector linear prediction (VLP) [10] was also employed. This technique uses a matrix of coefficients to predict the original frame of descriptors from the compressed frame. The VLP is calculated as [10]:

$$\text{VLP} = C_{01}(C_{00})^{-1},$$

where:

$$\begin{aligned} C_{01} &= \tfrac{1}{N}\sum_{n=0}^{N-1} x(n)\overline{x(n)}^T, \\ C_{00} &= \tfrac{1}{N}\sum_{n=0}^{N-1} x(n)x(n)^T. \end{aligned} \tag{4}$$

Table 13
Percentage incorrect frames for the prediction schemes, compared to no prediction

Descriptor	AP	AW	ASC	ASS	ASE	ASS/ASC	ALL
Non-predicted	92.7	72.2	89.7	90.3	0	37.5	4.1
VLP	93.1	87.4	96.8	92.9	0	63.1	26.2
Fixed pred	92.3	71.8	91.7	89.3	0	40	4.5

Table 14
SegSNR for prediction schemes, compared to no prediction

Descriptor	AP	AW	ASC	ASS	ASE	ASS/ASC	ALL
Non-predicted	39.6	33.5	36.9	46.2	30.2	39.5	38.2
VLP	35	28.1	31.7	38.6	25.7	33.2	32.5
Fixed pred	39.3	33.6	36.6	46	30.2	39.5	38.2

To augment the performance of Eqs. (3) and (4), both methods were utilized in conjunction with switched adaptive prediction [10]; a method that has been shown to operate well for quantisation of speech spectral envelop parameters [10]. This method trains classifiers for the compressed vectors offline using a vector quantiser (VQ) [11]. Separate predictors, Eqs. (3) and (4), are then trained for each classification. In the prediction process the input vector of descriptors is firstly classified (using the VQ) and then the original descriptors are estimated from the compressed descriptors, using the predictor that corresponds to the classification. For this paper the vectors were classified into 4 classes.

The search results using the proposed predictors in conjunction with the MP3 coder operating at 128 kbit/s are shown in Table 13 and the SegSNR results for the same test files are shown in Table 14. The results in Tables 13 and 14 represent utilizing the prediction schemes on test vectors not included in the training set. If vectors from within the training set were used, better results are achieved however, this would require training predictors for specific files. We are searching for a more generic solution.

The results in Tables 13 and 14 indicate that of the two prediction schemes, the simple fixed predictor achieved the best performance. As the VLP relies upon strong inter and intra-frame correlation, this result indicates that the inter and intra-frame correlation between the descriptor vectors is quite low. One could infer that the entropy of the descriptor vectors actually increases as the descriptors are grouped together, which supports the results of Section 2.8 where the search performance deteriorated when additional descriptors were added to the ASE to produce a meta-descriptor.

Whilst the fixed predictor achieved the best results of the two predictors, it achieved little or no improvement over using the non-predicted vectors. This result indicates that either the compression noise is non-stationary and thus not predictable, or alternately, that more sophisticated prediction techniques (non-linear, multi-tap) or sophisticated heuristic algorithms are required.

3.2. Machine learning techniques

A number of alternative approaches, emanating from the field of machine learning (ML) have been considered. Data mining is, in one aspect, a specialized application of machine learning, and as it essentially embodies much of the same key features, they are considered here as the same. The practical focus of these endeavours is to detect useful, but often-implicit patterns in empirical data, and to further construct descriptive models of these. Whilst there are several plausible techniques that could be considered in improving the performance of the low level descriptors, the impact of each is governed primarily by how the various aspects and concepts are both represented and modelled. A useful overall guide to this technology is found in [12].

The approach taken in this paper is to learn rule-based predictive models, based on compression-affected descriptors that can essentially predict the former compression free state. An overview of this process is shown in Fig. 1.

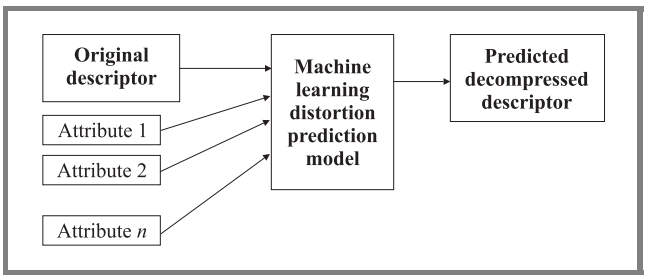

Fig. 1. Overview of ML techniques.

Several trials were undertaken to develop a range of non-parametric regression models that would map these

descriptors. As such, all of the available descriptors on one side, say those derived from the compressed file, are used to form a prediction model for each of the non-effected descriptors. Thus this technique can be viewed as being closest to that of the VQ method of the previous section. However, despite the availability of employing all descriptors to build a suitable predictor, the machine learning algorithm, Cubist [13], used only a small number of these in each case (selected via various entropy-based metrics).

The results shown in Table 15 compare the search performance for the meta-descriptor comprising all of the descriptors when using, ML prediction, the ASP predictors of Section 3.1 and no prediction. The meta-descriptor was selected to evaluate the performance of the ML predictor, as the 37 dimensional input and output vectors present the most challenging scenario. As can be seen in Table 15, the ML approach produces the most reliable search performance. The ML predictor significantly improves the missed-framing rate by 0.9% compared to the simple non-predicted method. This actually represents a 22% improvement in searching performance.

Table 15
Comparative improved error rate in matching frames

Descriptor	ALL
Non-predicted	4.1
VLP	26.2
Fixed pred	4.5
ML: Cubist	3.2

This improvement is readily explained by the nature of the machine learning approach. As such algorithms seek to learn specific concepts, they create and grow a model to fit or explain the training data. In contrast, traditional signal processing techniques often apply the reverse situation, where a fixed model is employed and data fitted to the model structure. Under the ML regime, several virtual contexts are formed and utilized, compared to the VQ predictor of the last section that used 4.

The type of ML algorithm used here, produces what could be best described as a combined decision tree with a series of linear regression models, these reside at the leaf nodes of the decision tree. The internal nodes and branch structure assist in segmenting the descriptor data into several regions of similar magnitudes (at the leaves) where afterwards, suitable linear regression representations finally model any residual variance within each leaf.

Whilst the ML prediction provides a significant improvement in search performance, the reliability of the search is still not sufficiently high for practical applications (a failure rate of approximately 0.1% may be deemed necessary). To improve the overall searching reliability a more sophisticated searching algorithm and ML predictor are proposed and analysed in Section 3.3.

3.3. Improved search algorithm and ML predictor

This section proposes an extension to the simple MSE searching method proposed in Section 2.2. The new method is still based on MSE minimization but now employs the calculation across the previous, current and future (PCF) frames. Incorporating a larger time scale (adjacent frames) into the calculation allows more accurate searching, as the evolution, pattern or trend is identified as opposed to the previous method of matching only a single value. An example of this improved matching performance is illustrated in Fig. 2. If we consider the nine samples illustrated in Fig. 2 as separate individual samples, and, if the target sample is that second from the left, it is easy to visualise how a number of the other samples could be incorrectly located in the presence of ambiguous noise. However, when groups of three consecutive samples are used, it is easy to identify that the smooth curve represented by the left-hand group is easily distinguished from the other groups.

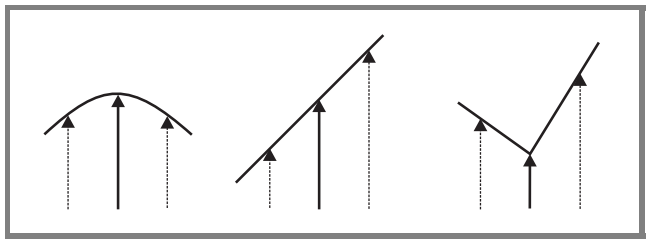

Fig. 2. Examples trends searched for in descriptor data using the PCF MSE method.

The PCF method essentially provides a localised context that increases the frame's resolution by a factor of three. Therefore singular dimensioned descriptors (AP, ASC, ASS) are now 3 dimensions wide, AW is 6 dimensions wide and the ASE descriptor now has an effective resolution of 96. The trade off of the proposed PCF scheme is that the temporal resolution of the search is reduced. By extending the MSE method, the algorithm remains consistent with those discussed in [8, 9].

The frame that minimizes Eq. (5) is considered the closest matched frame for the PCF:

$$\mathrm{PCF}_{\mathrm{MSE}} = \frac{1}{3r}\left(\begin{array}{l}\sum_{n=1}^{r}(x_{pn}-\overline{x}_{pn})^2+\sum_{n=1}^{r}(x_{cn}-\overline{x}_{cn})^2+\\ \sum_{n=1}^{r}(x_{fn}-\overline{x}_{fn})^2\end{array}\right), \tag{5}$$

where x_n, $\overline{x_n}$ and r are as defined for Eq. 2 and the additional p, c and f stand for the previous, current and future frames respectively. Table 16 displays an average representation for the percentage of unsuccessfully matched frames using MP3 – 128 kbit/s encoding.

Comparing the results in Table 16 to Tables 2, 4, 6, 8 and 10, indicates that adding the previous and future frames into the search criteria, results in a dramatic improvement

for all descriptors. Even the least reliable descriptor for searching compressed files, ASS, has its incorrect matches reduced from 90.6% to 15.7%. The ASE descriptor now produces incorrect matches on 0.4% of occasions. This is approximately a third of the error rate of the simple search and is approaching our target of 0.1%. However, the trade off for the improved performance is increased complexity and reduced temporal resolution.

Table 16
Percentage incorrect frames for the PCF search

	AP	AW	ASC	ASS	ASE
Inst.	10.3	4.8	10.4	13.9	0.4
Comb.	9.3	4.8	13.3	17.4	0.4
Average	9.8	4.8	11.9	15.7	0.4

We subsequently modified our ML algorithm to additionally incorporate the past and future frames into the range of attributes available for building the various prediction models. This resulted in a modest improvement in search reliability of approximately 0.3% for the single value descriptor AP. Whilst this improvement in performance may be able to be translated to the ASE descriptor, this was not tested due to the dramatic increase in complexity and storage required to build predictors for each of the 32 ASE elements.

4. Conclusion

An extended analysis of lossy compression effects on the MPEG-7 low level audio descriptors was conducted. This analysis exposed a distinct degradation in the performance of simple practical searching schemes when lossy compression has been used to modify the MM files. Methods to reduce the effects of compression in practical searching were then investigated. Of the proposed methods, prediction schemes based on machine learning were found to offer the greatest reduction in distortion. However, these prediction schemes did not completely nullify the effect of compression. A more sophisticated search mechanism that employs multiple frames in its calculation was then proposed. Generally this scheme significantly improved the reliability of searching, however, even the best performing descriptor combination still did not provide adequate performance.

A possible method for improving the search performance of the MPEG-7 audio descriptors with compressed input data would be to develop new compression algorithms that maintain the integrity of at least some of the descriptors. However, due to the prevalence of existing audio compression schemes (such as MP3) it is highly unlikely that such a new compression algorithm would be widely adopted. The authors instead propose that future work should focus on developing better search mechanisms such as those based on maximum likelihood. In addition and more importantly, new audio descriptors that incorporate the characteristics of existing compression schemes into their structure should be developed.

References

[1] ISO/IEC JTC1/SC29/WG11/N4031, "Overview of the MPEG-7 Standard (version 5)", International Organisation for Standardisation, Singapore, March 2001.

[2] S. Chang, T. Sikora, and A. Puri, "Overview of the MPEG-7 standard", *IEEE Trans. Circ. Syst. Video Technol.*, vol. 11, no. 6, pp. 688–695, 2001.

[3] ISO/IEC FDIS 15938-4, "Information technology multimedia content description interface", Part 4: "Audio", International Organisation for Standardisation, Singapore, March 2001.

[4] M. Casey, "MPEG-7 sound recognition tools", *IEEE Trans. Circ. Syst. Video Technol.*, vol. 11, no. 6, pp. 737–747, 2001.

[5] M. Hu and Y. Jian, "MD2L: content description of multimedia documents for efficient process and search/retrieval", in *Proc. IEEE Forum Res. & Technol. Adv. Dig. Libr.*, 1999, pp. 200–213.

[6] ISO/IEC JTCI/SC29, "Information technology-coding of motion pictures and associated audio for digital storage media upto about 1.5 Mbit/s – IS 11172", Part 3: "Audio", 1992.

[7] Microsoft, "Windows media encoder", July 2002, available at http://www.microsoft.com/windows/windowsmedia/WM7/encoder/whitepaper.asp

[8] S. Quackenbush and A. Lindsay, "Overview of MPEG-7 audio", *IEEE Trans. Circ. Syst. Video Technol.*, vol. 11, no. 6, pp. 725–729, 2001.

[9] E. Allamanche, "Robust matching of audio signals using spectral flatness features", in *IEEE Worksh. Appl. Signal Proc. Audio Acoust.*, 2001, pp. 127–130.

[10] M. Yong, G. Davidson, and A. Gersho, "Encoding of LPC spectral parameters using switched-adaptive interframe vector prediction", in *Proc. ICASSP*, 1988, vol. 1, pp. 402–405.

[11] A. Gersho and R. M. Gray, *Vector Quantisation and Signal Compression*. Kluwer, 1992.

[12] I. H. Witten and E. Frank, *Data Mining Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, 2000.

[13] Cubist (version 1.13), Rulequest Research, www.rulequest.com

Jason Lukasiak is currently a lecturer in the School of Electrical, Computer and Telecommunications Engineering at the University of Wollongong. He received his B.E. (Hons.) and Ph.D. from the University of Wollongong in 1998 and 2002, respectively. Prior to commencing his Ph.D. studies in 1999, he was employed by BHP steel from 1987. During his employement with BHP his positions ranged from computer network technician to electrical project engineer. The topic of his Ph.D. thesis was "Techniques for low-rate scalable speech compression" and other current research interests include description and adaptation of multimedia objects, linked audio-visual modeling and transcoding of speech signals. In relation to the

MPEG-7 work detailed in this paper, a web based implementation allowing calculation of the MPEG-7 low level audio descriptors for any input audio file has been completed.

e-mail: jasonl@elec.uow.edu.au
School of Electrical, Computer
and Telecommunications Engineering
University of Wollongong
Wollongong, NSW 2522, Australia

David Stirling has developed considerable expertise in data analysis and knowledge management with skills in problem solving, statistical methods, visualization, pattern recognition, data fusion and reduction, and programming and is widely experienced in applying these to organizations requiring solutions to complex concept relationship problems. He has applied machine learning and data mining techniques of specialised classifier designs for noisy multivariate data to medical research, exploration geoscience, and financial markets, as well as to industrial primary operations. Before setting up his own consultancy company in 1998, Dr. Stirling was a principal research scientist with BHP research (and before that, with John Lysaght) and has over 15 years experience working in the Port Kembla steelworks. He has recently taken up a position as senior lecturer in the School of Electrical, Computer and Telecommunications Engineering at the University of Wollongong.

e-mail: stirling@uow.edu.au
School of Electrical, Computer
and Telecommunications Engineering
University of Wollongong
Wollongong, NSW 2522, Australia

Shane Perrow graduated from the University of Wollongong with a B.E. (Hons.) in December 2002. The work contributed by Shane to the paper formed the majority of his final year honours thesis topic.

e-mail: Perrow@ali.com.au
School of Electrical, Computer
and Telecommunications Engineering
University of Wollongong
Wollongong, NSW 2522, Australia

Nick Harders is a final year undergraduate student in maths/electrical engineering at the University of Wollongong. He completed his contribution to the paper whilst working on a summer university scholarship program.

e-mail: nharders@bigpond.com
School of Electrical, Computer
and Telecommunications Engineering
University of Wollongong
Wollongong, NSW 2522, Australia